Procesamiento del Lenguaje Natural, nº39 (2007), pp. 197-204 recibido 30-04-2007; aceptado 22-06-2007

Cognitive Modules of an NLP Knowledge Base for Language Understanding

Carlos Periñán-Pascual
Universidad Católica San Antonio
Campus de los Jerónimos s/n
30107 Guadalupe - Murcia (Spain)
jcperinan@pdi.ucam.edu

Francisco Arcas-Túnez
Universidad Católica San Antonio
Campus de los Jerónimos s/n
30107 Guadalupe - Murcia (Spain)
farcas@pdi.ucam.edu

Resumen: Algunas aplicaciones del procesamiento del lenguaje natural, p.ej. la traducción automática, requieren una base de conocimiento provista de representaciones conceptuales que puedan reflejar la estructura del sistema cognitivo del ser humano. En cambio, tareas como la indización automática o la extracción de información pueden ser realizadas con una semántica superficial. De todos modos, la construcción de una base de conocimiento robusta garantiza su reutilización en la mayoría de las tareas del procesamiento del lenguaje natural. El propósito de este artículo es describir los principales módulos cognitivos de FunGramKB, una base de conocimiento léxico-conceptual multipropósito para su implementación en sistemas del procesamiento del lenguaje natural.
Palabras clave: Representación del conocimiento, ontología, razonamiento, postulado de significado.

Abstract: Some natural language processing systems, e.g. machine translation, require a knowledge base with conceptual representations reflecting the structure of human beings' cognitive system. In some other systems, e.g. automatic indexing or information extraction, surface semantics could be sufficient, but the construction of a robust knowledge base guarantees its use in most natural language processing tasks, consolidating thus the concept of resource reuse. The objective of this paper is to describe FunGramKB, a multipurpose lexico-conceptual knowledge base for natural language processing systems. Particular attention will be paid to the two main cognitive modules, i.e. the ontology and the cognicon.
Keywords: Knowledge representation, ontology, reasoning, meaning postulate.

1 FunGramKB

FunGramKB Suite[1] is a user-friendly environment for the semiautomatic construction of a multipurpose lexico-conceptual knowledge base for a natural language processing (NLP) system within the theoretical model of S.C. Dik's Functional Grammar (1978, 1989, 1997). FunGramKB is not a literal implementation of Dik's lexical database, but we depart from the functional model in some important aspects with the aim of building a more robust knowledge base.

On the one hand, FunGramKB is multipurpose in the sense that it is both multifunctional and multilanguage. In other words, FunGramKB can be reused in various NLP tasks (e.g. information retrieval and extraction, machine translation, dialogue-based systems, etc) and with several natural languages.[2]

[1] We use the name 'FunGramKB Suite' to refer to our knowledge engineering tool and 'FunGramKB' to the resulting knowledge base.

[2] English, Spanish, German, French and Italian are supported in the current version of FunGramKB.

ISSN: 1135-5948

On the other hand, our knowledge base is lexico-conceptual, because it comprises two general levels of information: a lexical level and a cognitive level. In turn, these levels are made up of several independent but interrelated components:

Lexical level (i.e. linguistic knowledge):
- The lexicon stores morphosyntactic, pragmatic and collocational information of words.
- The morphicon helps our system to handle cases of inflectional morphology.

Cognitive level (i.e. non-linguistic knowledge):
- The ontology is presented as a hierarchical structure of all the concepts that a person has in mind when talking about everyday situations.
- The cognicon stores procedural knowledge by means of cognitive macrostructures, i.e. script-like schemata in which a sequence of stereotypical actions is organised on the basis of temporal continuity, and more particularly on James Allen's temporal model (Allen, 1983, 1991; Allen and Ferguson, 1994).
- The onomasticon stores information about instances of entities, such as people, cities, products, etc.

The main consequence of this two-level design is that every lexical module is language-dependent, while every cognitive module is shared by all languages. In other words, computational lexicographers must develop one lexicon and one morphicon for English, one lexicon and one morphicon for Spanish and so on, but knowledge engineers build just one ontology, one cognicon and one onomasticon to process any language input cognitively. Section 2 gives a brief account on the psychological foundation of FunGramKB cognitive level, and sections 3 and 4 describe the two main cognitive modules in that level, i.e. the ontology and the cognicon.

2 Cognitive knowledge in natural language understanding

In cognitive psychology, common-sense knowledge is usually divided into three different types (Tulving, 1985):

- Semantic knowledge, which stores cognitive information about words; it is a kind of mental dictionary.
- Procedural knowledge, which stores information about how events are performed in ordinary situations—e.g. how to ride a bicycle, how to fry an egg...; it is a kind of manual for everyday actions.
- Episodic knowledge, which stores information about specific biographic events or situations—e.g. our wedding-day; it is a kind of personal scrapbook.

Therefore, if there are three types of knowledge involved in human reasoning, there must be three different kinds of knowledge schemata. These schemata are successfully mapped in an integrated way into the cognitive component of FunGramKB:

- Semantic knowledge is represented in the form of meaning postulates in the ontology.
- Procedural knowledge is represented in the form of cognitive macrostructures in the cognicon.
- Episodic knowledge can be stored as a case base.[3]

A key factor for successful reasoning is that all these knowledge schemata (i.e. meaning postulates, cognitive macrostructures and cases) must be represented through the same formal language, so that information sharing could take place effectively among all cognitive modules. Our formal language is partially founded on Dik's model of semantic representation (1978, 1989, 1997), which was initially devised for machine translation (Connolly and Dik, 1989).

Computationally speaking, when storing cognitive knowledge through FunGramKB Suite, a syntactic-semantic checker is triggered, so that consistent well-formed constructs can be stored. Moreover, a parser outputs an XML-formatted feature-value structure used as the input for the reasoning engine, so that

[3] FunGramKB can be very useful in case-based reasoning, where problems are solved by remembering previous similar cases and reusing general knowledge.

inheritance and inference mechanisms can be applied. Both the syntactic-semantic validator of meaning postulates and the XML parser were written in C#.

3 FunGramKB ontology

Nowadays there is no single right methodology for ontology development. Ontology design tends to be a creative process, so it is probable that two ontologies designed by different people have a different structuring (Noy and McGuinness, 2001). To avoid this problem, the ontology model should be founded on a solid methodology. The remaining of this section describes five methodological criteria applied to FunGramKB ontology, some of which are based on principles implemented in other NLP projects (Bouaud et al., 1995; Mahesh, 1996; Noy and McGuinness, 2001). The definition of these criteria in the analysis and design phases of the ontology model and the strict application of these guidelines in the development phase contributed to avoid some common errors in conceptual modelling.

3.1 Symbiosis between universality and linguistic motivation

FunGramKB ontology takes the form of a universal concept taxonomy, where ‘universal’ means that every concept we can imagine has an appropriate place in this ontology. On the other hand, our ontology is linguistically motivated, as a result of its involvement with the semantics of lexical units, although the knowledge stored in our ontology is not specific to any particular language.

3.2 Subsumption as the only taxonomic relation

At first sight, it can seem that the exclusive use of the IS-A relation can impoverish the ontological model. Indeed, a consequence of this restriction on the taxonomic relation is found in the modelling of the upper level, where metaconcepts #ENTITY, #EVENT and #QUALITY arrange nouns, verbs and adjectives respectively in cognitive dimensions. However, the fact that concepts linked to lexical units of different grammatical categories are not explicitly connected in our ontological model doesn’t prevent FunGramKB to relate those lexical units in the cognitive level through their meaning postulates. Indeed, our ontology establishes a high degree of connectivity among conceptual units by taking into account semantic components which are shared by their meaning postulates. In order to incorporate human beings’ commonsense, our ontology must identify the relations which can be established among conceptual units, and hence among lexical units. However, displaying semantic similarities and differences through taxonomic relations turns out to be more chaotic than through meaning postulates linked to conceptual units.

3.3 Three-layered ontological model

FunGramKB ontology distinguishes three different conceptual levels, each one of them with concepts of a different type: metaconcepts, basic concepts and terminals. Figure (1) illustrates these three types of concepts.

```
#ENTITY
  → #PHYSICAL
    → #OBJECT
      → #SELF_CONNECTED_OBJECT
        → +ARTIFICIAL_OBJECT
          → +CORPUSCULAR
            → +SOLID
              → +BALL
                → $FOOTBALL
```

Figure 1: Example of ontological structuring in FunGramKB

Metaconcepts, preceded by symbol #, constitute the upper level in the taxonomy. The analysis of the upper level in the main linguistic ontologies—DOLCE (Gangemi et al., 2002; Masolo et al., 2003), Generalized Upper Model (Bateman, 1990; Bateman, Henschel and Rinaldi, 1995), Mikrokosmos (Beale, Nirenburg and Mahesh, 1995; Mahesh and Nirenburg, 1995; Nirenburg et al., 1996), SIMPLE (Lenci, 2000; Lenci et al., 2000; Pedersen and Keson, 1999; SIMPLE Specification Group, 2000; Villegas and Brosa, 1999), SUMO (Niles and Pease, 2001a, 2001b)—led to a metaconceptual model whose design contributes to the integration and exchange of information with other ontologies, providing thus standardization and uniformity. Since metaconcepts reflect cognitive dimensions, they are not assigned meaning postulates. Therefore, our metaconcepts play the role of 'hidden categories', i.e. concepts which aren't linked to any lexical unit so that they can serve as hidden superordinates and avoid circularity.

Basic concepts, preceded by symbol +, are used in FunGramKB as defining units which enable the construction of meaning postulates for basic concepts and terminals, as well as taking part as selection preferences in thematic frames. The starting point for the identification of basic concepts was the defining vocabulary in *Longman Dictionary of Contemporary English* (Procter, 1978), though deep revision was required in order to perform cognitive mapping.

Finally, terminals are headed by symbol $. The borderline between basic concepts and terminals is based on their definitory potential to take part in meaning postulates.

3.4 Non-atomicity of conceptual units

In FunGramKB, basic and terminal concepts are not stored as atomic symbols but are provided with a rich internal structure consisting of semantic properties such as the thematic frame or the meaning postulate.

On the one hand, every event in the ontology is assigned one thematic frame, i.e. a prototypical cognitive construct which states the number and type of participants involved in the cognitive situation portrayed by the event. In turn, predicate frames of verbs in the lexicon are constructed from thematic frames in the ontology. For instance, *hundir* and *zozobrar* are Spanish verbs which trigger the same thematic frame, since both of them are linked to the same concept (example 1).

(1) SINK $(x_1)_{Agent}$ $(x_2)_{Theme}$ $(x_3$: LIQUID ^ MUD$)_{Location}$ $(x_4)_{Origin}$ $(x_5)_{Goal}$ $(f_1$: SLOW$)_{Speed}$

However, these verbs can differ in their predicate frames, since they show different profiled arguments (examples 2-3).

(2) hundir $(x_1)_{NP / S / Agent}$ $(x_2)_{NP / DO / Theme}$
hundir $(x_2)_{NP / S / Theme}$

(3) zozobrar $(x_2)_{NP / S / Theme}$

In other words, these lexical units are linked to the same thematic frame at the cognitive level, but the instantiation of this thematic frame can make divergences occur in predicate frames at the lexical level.[4]

On the other hand, a meaning postulate is a set of one or more logically connected predications (e_1, e_2... e_n), which are cognitive constructs carrying the generic features of the concept.[5] Concepts, and not words, are the building blocks for the formal description of meaning postulates, so a meaning postulate becomes a language-independent semantic knowledge representation. To illustrate, some predications in the meaning postulates of an entity, event and quality are presented in examples (4), (5) and (6) respectively:[6]

(4) BIRD
+(e_1: BE $(x_1$: BIRD$)_{Theme}$ $(x_2$:VERTEBRATE$)_{Referent}$)
*(e_2: HAVE $(x_1)_{Theme}$ $(x_3$: $_m$ FEATHER $_{\&\ 2}$ LEG $_{\&\ 2}$ WING$)_{Referent}$)
*(e_3: FLY $(x_1)_{Theme}$)

(5) KISS
+(e_1: TOUCH $(x_1$: PERSON$)_{Agent}$ $(x_2)_{Theme}$ $(f_1$: $_2$ LIP$)_{Instrument}$ $(f_2$: (e_2: LOVE $(x_1)_{Agent}$ $(x_2)_{Theme}$) | (e_2: GREET $(x_1)_{Agent}$ $(x_2)_{Theme}$))$_{Reason}$)

(6) HUGE
+(e_1: BE $(x_2)_{Theme}$ $(x_1$: HUGE$)_{Attribute}$)
+(e_2: BE $(x_1)_{Theme}$ $(x_3$: SIZE$)_{Referent}$)
+(e_3: BE $(x_2)_{Theme}$ $(x_4$: $_m$ BIG$)_{Attribute}$)

For instance, predications in example (1) have the following natural language equivalents:

Birds are always vertebrates.
A typical bird has many feathers, two legs and two wings.
A typical bird flies.

Dik (1997) proposes using words from the own language when describing meaning postulates, since meaning definition is an internal issue of the language. However, this strategy contributes to lexical ambiguity due to the polysemic nature of the defining lexical

---

[4] The difference between thematic frames and predicate frames is partially grounded on the distinction between argument roles and participant roles in Goldberg's Construction Grammar (1995).

[5] Periñán Pascual and Arcas Túnez (2004) describe the formal grammar of well-formed predications for meaning postulates in FunGramKB.

[6] For the sake of clarity, the names of conceptual units have been oversimplified.

units. In addition, describing the meaning of words in terms of other words leads to some linguistic dependency (Vossen, 1994). Instead, FunGramKB employs concepts for the formal description of meaning postulates, resulting in an interlanguage representation of meaning.

An alternative could have been to use second-order predicate logics for the formal representation of lexical meaning. However, the problem lies not only on the little expressive power of predicate logics, but also on the fact that standard logics use monotonic reasoning, which isn't robust enough for the simulation of human beings' commonsense reasoning.

3.5 Meaning postulates as ontological organizers

Our ontology structuring complies with the similarity, specificity and opposition principles applied to the meaning postulates of concepts. Firstly, all subordinate concepts must share the meaning postulate of their superordinate concept (i.e. similarity principle). Secondly, all subordinate concepts must have a meaning postulate which states a distinctive feature (or *differentiae*) not present in the meaning postulate of its superordinate concept (i.e. specificity principle). Finally, *differentiae* in the meaning postulates of sibling concepts must be incompatible one another (i.e. opposition principle).

4 FunGramKB cognicon

Text understanding must not be restricted to the comprehension of individual sentences, but it must involve the integration of all this information into a 'situation model' (Zwaan and Radvansky, 1998) with the purpose of reconstructing the textual world underlying to the literal sense of the linguistic realizations which make up the text surface. The task of reconstructing the situation model of an input text requires NLP systems to hold human beings' commonsense knowledge in the form of generic cognitive structures which can facilitate inferences and predictions as well as information selection and management. Since scripts were devised by Schank and Abelson (1977), little effort has been made to build a large-scale database of procedural-knowledge schemata. For example, both expectation packages (Gordon, 1999) and *ThoughtTreasure* (Mueller, 1999) are systems which contain facts and rules about ordinary situations, but it is very difficult to apply any case-based reasoning on them.

In FunGramKB, meaning postulates are not sufficient to describe commonsense knowledge, but they contribute actively to build 'cognitive macrostructures' in the cognicon. In other words, our knowledge base integrates semantic knowledge from the ontology with procedural knowledge from the cognicon, resulting in a correlation that almost no NLP system has achieved yet. These schemata are described as 'macrostructures' because they are more comprehensive constructions than meaning postulates. While meaning postulates are ontology-oriented knowledge representations, cognitive macrostructures organize knowledge in scenes according to temporality and causality parameters. On the other hand, these macrostructures are described as 'cognitive' because they are built with conceptual units from the ontology. Unlike most natural language understanding systems, expectations about what is about to happen in a particular situation are not lexical but conceptual, so different lexical realizations with the same meaning in the same or different languages correspond to the same expectation in FunGramKB.

In example (7), we present some predications of the cognitive macrostructure *Eating_at_restaurants*:

(7) (e_1: ENTER (x_1: CUSTOMER)$_{\text{Theme}}$ (x_2: RESTAURANT)$_{\text{Goal}}$ (f_1: (e_2: BE (x_1) (x_3: HUNGRY)$_{\text{Attrribute}}$))$_{\text{Reason}}$)
(e_3: ACCOMPANY (x_4: WAITER)$_{\text{Theme}}$ (x_1)$_{\text{Referent}}$ (f_2: TABLE)$_{\text{Goal}}$)
(e_4: SIT (x_1)$_{\text{Theme}}$ (x_5: f_1)$_{\text{Location}}$)
(e_5: BRING (x_4)$_{\text{Theme}}$ (x_6: MENU ^ WINE_LIST)$_{\text{Referent}}$ (f_3: x_1)$_{\text{Goal}}$)
(e_6: REQUEST (x_1)$_{\text{Theme}}$ (x_7: FOOD | BEVERAGE)$_{\text{Referent}}$ (x_4)$_{\text{Goal}}$)
(e_7: TELL (x_4)$_{\text{Theme}}$ (x_8: (e_8: COOK (x_9: COOK)$_{\text{Theme}}$ (x_{10}: FOOD)$_{\text{Referent}}$)$_{\text{Referent}}$ (x_9)$_{\text{Goal}}$)
(e_9: BRING (x_4)$_{\text{Theme}}$ (x_{11}: BEVERAGE)$_{\text{Referent}}$ (f_4: BAR)$_{\text{Source}}$)

The main advantage of this approach is that meaning postulates and cognitive macrostructures are represented through the same formal language, so that knowledge can be shared more effectively between FunGramKB cognitive modules, particularly when reasoning mechanisms are triggered.

5 Reasoning engine in FunGramKB

An NLP application is actually a knowledge-based system, so it must be provided with a knowledge base and a reasoning engine. Two reasoning processes have been devised to work with FunGramKB cognitive modules: MicroKnowing and MacroKnowing.

MicroKnowing (Microconceptual-Knowledge Spreading) is a multi-level process performed by means of two types of reasoning mechanisms: inheritance and inference. Our inheritance mechanism strictly involves the transfer of one or several predications from a superordinate concept to a subordinate one in the ontology. On the other hand, our inference mechanism is based on the structures shared between predications linked to conceptual units which do not take part in the same subsumption relation within the ontology. Cyclical application of the inheritance and inference mechanisms on our meaning postulates allow FunGramKB to minimize redundancy as well as keeping our knowledge base as informative as possible. When the language engineer modifies an existing meaning postulate or builds a new one, just before being stored, FunGramKB Suite automatically performs the MicroKnowing for that meaning postulate in order to check the compatibility of the newly-incorporated predications with other predications involved in the reasoning process. The language engineer is informed about any incompatibility with inferred or inherited predications. In addition, FunGramKB Suite displays the whole MicroKnowing process step by step, enabling us to verify inference and inheritance conditions in a transparent way.[7]

Currently we are working on the MacroKnowing (Macroconceptual-Knowing Spreading), i.e. the process of integrating meaning postulates from the ontology with the cognitive macrostructures in the cognicon in order to spread the procedural knowledge stored in FunGramKB. This interaction of semantic and procedural knowledge, so distinctive of human reasoning, is hardly found in NLP systems to date.

[7] Periñán Pascual and Arcas Túnez (2005) give an accurate description of MicroKnowing in FunGramKB.

6 Conclusion

In NLP, knowledge is usually applied to the input text for two main tasks: parsing (e.g. spell checking, syntactic ambiguity resolution, etc) and partial understanding (e.g. lexical ambiguity resolution, document classification, etc). Full natural language understanding is hardly performed. Indeed, deep semantics for NLP is currently very limited, perhaps because most applications exploit WordNet as a source of information. Moreover, researchers do not even agree on how much semantic information is sufficient to achieve the best outcome. However, it is thought that performance is improved if the system is provided with a robust knowledge base and a powerful inference component (Vossen, 2003). In fact, the main problem in the successful development of natural language understanding systems lies on the lack of an extensive commonsense knowledge base. Since commonsense is mainly made up of semantic and procedural knowledge, which FunGramKB stores in the form of meaning postulates and cognitive macrostructures respectively, we can conclude that FunGramKB can help language engineers to design more intelligent NLP applications.

Bibliography

Allen, J. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM,* 26 (11): 832-843.

Allen, J. 1991. Time and time again: the many ways to represent time. *International Journal of Intelligent Systems,* 6 (4): 341-355.

Allen, J. and G. Ferguson. 1994. Actions and events in interval temporal logic. *Journal of Logic and Computation,* 4 (5): 531-579.

Bateman, J.A. 1990. Upper modeling: a general organization of knowledge for natural language processing. *Workshop on Standards for Knowledge Representation Systems*. Santa Barbara.

Bateman, J.A., R. Henschel, and F. Rinaldi. 1995. The Generalized Upper Model 2.0. Technical report. IPSI/GMD, Darmstadt.

Beale, S., S. Nirenburg, and K. Mahesh. 1995. Semantic analysis in the Mikrokosmos machine translation project. *Proceedings of* *the Symposium on NLP*. Bangkok.

Bouaud, J., B. Bachimont, J. Charlet, and P. Zweigenbaum. 1995. Methodological principles for structuring an ontology. *Proceedings of IJCAI'95: Workshop on Basic Ontological Issues in Knowledge Sharing*. Montreal.

Connolly, J.H. and S.C. Dik. eds. 1989. *Functional Grammar and the Computer*. Foris, Dordrecht.

Dik, S.C. 1978. *Functional Grammar*. Foris, Dordrecht.

Dik, S.C. 1989. *The Theory of Functional Grammar*. Foris, Dordrecht.

Dik, S.C. 1997. *The Theory of Functional Grammar*. Mouton de Gruyter, Berlin-New York.

Gangemi, A., N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. 2002. Sweetening ontologies with DOLCE. *Proceedings of EKAW 2002*. 166-181, Sigüenza.

Goldberg, A.E. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. The University of Chicago Press, Chicago.

Gordon, A.S. 1999. The design of knowledge-rich browsing interfaces for retrieval in digital libraries. Doctorate thesis. Northwestern University.

Lenci, A. 2000. Building an ontology for the lexicon: semantic types and word meaning. *Workshop on Ontology-Based Interpretation of Noun Phrases*. Kolding.

Lenci, A., N. Bel, F. Busa, N. Calzolari, E. Gola, M. Monachini, A. Ogonowski, I. Peters, W. Peters, N. Ruimy, M. Villegas, and A. Zampolli. 2000. SIMPLE: a general framework for the development of multilingual lexicons. *International Journal of Lexicography,* 13 (4): 249-263.

Mahesh, K. 1996. Ontology development for machine translation: ideology and methodology. Technical report MCCS-96-292. Computing Research Laboratory, New Mexico State University, Las Cruces.

Mahesh, K. and S. Nirenburg. 1995. Semantic classification for practical natural language processing. *Proceedings of the 6th ASIS SIG/CR Classification Research Workshop: An Interdisciplinary Meeting*. 79-94, Chicago.

Masolo, C., S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. 2003. WonderWeb deliverable D18: ontology library. Technical report. Laboratory for Applied Ontology, ISTC-CNR.

Mueller, E.T. 1999. A database and lexicon of scripts for ThoughtTreasure. [http://cogprints.ecs.soton.ac.uk/archive/00000555/]

Niles, I. and A. Pease. 2001a. Origins of the Standard Upper Merged Ontology: a proposal for the IEEE Standard Upper Ontology. *Working Notes of the IJCAI-2001 Workshop on the IEEE Standard Upper Ontology*. Seattle.

Niles, I. and A. Pease. 2001b. Towards a standard upper ontology. *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems*. Ogunquit.

Nirenburg, S., S. Beale, K. Mahesh, B. Onyshkevych, V. Raskin, E. Viegas, Y. Wilks, and R. Zajac. 1996. Lexicons in the MikroKosmos project. *Proceedings of the AISB'96 Workshop on Multilinguality in the Lexicon*. Brighton.

Noy, N.F. and D.L. McGuinness. 2001. Ontology development 101: a guide to creating your first ontology. Technical report KSL-01-05. Stanford Knowledge Systems Laboratory, Stanford University.

Pedersen, B.S. and B. Keson. 1999. SIMPLE—semantic information for multifunctional plurilingual lexica: some examples of Danish concrete nouns. *Proceedings of the SIGLEX-99 Workshop*. Maryland.

Periñán Pascual, C. and F. Arcas Túnez. 2004. Meaning postulates in a lexico-conceptual knowledge base. *Proceedings of the 15th International Workshop on Databases and Expert Systems Applications*. 38-42, IEEE, Los Alamitos.

Periñán Pascual, C. and F. Arcas Túnez. 2005. Microconceptual-Knowledge Spreading in FunGramKB. *Proceedings on the 9th IASTED International Conference on Artificial Intelligence and Soft Computing*. 239-244, ACTA Press, Anaheim-Calgary-Zurich.

Procter, P. ed. 1978. *Longman Dictionary of Contemporary English*. Longman, Harlow.

Schank, R. and R.P. Abelson. 1977. *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum, Hillsdale.

SIMPLE Specification Group. 2000. Specification SIMPLE Work Package 2: linguistic specifications deliverable D2.1. Technical report.

Tulving, E. 1985. How many memory systems are there? *American Psychologist,* 40: 385-398.

Villegas, M. and I. Brosa. 1999. Spanish SIMPLE: lexicon documentation. Technical report.

Vossen, P. 1994. The end of the chain: where does decomposition of lexical knowledge lead us eventually? E. Engberg-Pedersen, L. Falster Jakobsen, and L. Schack Rasmussen. eds. *Function and Expression in Functional Grammar*. 11-39, Mouton de Gruyter, Berlin-New York.

Vossen, P. 2003. Ontologies. R. Mitkov. ed. *The Oxford Handbook of Computational Linguistics*. 464-482, Oxford University Press, Oxford.

Zwaan, R.A. and G.A. Radvansky. 1998. Situation models in language comprehension and memory. *Psychological Bulletin,* 123 (2): 162-185.